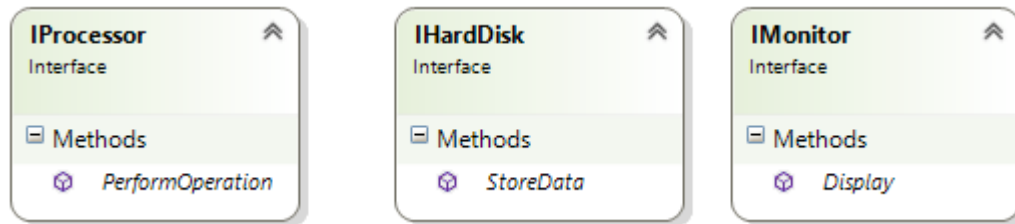


## Assignment 1 ('Factory Method')

In this assignment computers are created in computershops; your task is to use Design Pattern 'Factory Method' for creating these computers. This means that computer parts (processor, monitor and hard disk) are being created through virtual or abstract methods (in the computershop). There are two different shops, a LowBudgetShop creating cheap computer parts, and a HighBudgetShop creating expensive computer parts.

Use the following interfaces for the computer parts:



The computer parts are created in (factory) method "AssembleMachine" of the ComputerShop class, and for each part, the corresponding method is being called (e.g. "PerformOperation" of the processor).

Use the Main code below:

```
static void Main(string[] args)
{
    // create a shop where they assemble expensive computers
    Console.WriteLine("[shop creating expensive computers]");
    // ... create shop
    // ... assemble (one) machine

    Console.WriteLine();

    // create a shop where they assemble cheap computers
    Console.WriteLine("[shop creating cheap computers]");
    // ... create shop
    // ... assemble (one) machine

    Console.ReadKey();
}
```

to generate the output to the right:

The screenshot shows a console window with the following output:

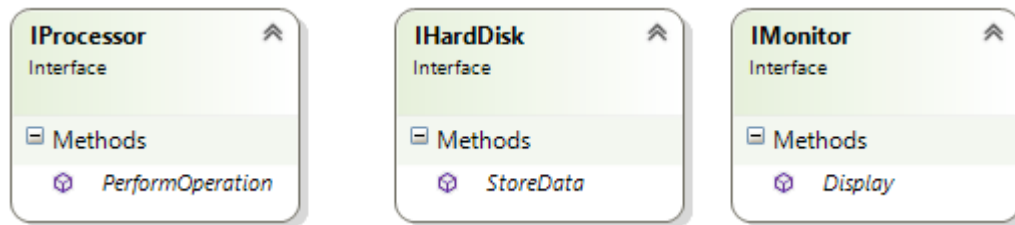
```
file:///C:/Users/Gerwin van Dijken...
[shop creating expensive computers]
performing operation very quickly...
storing data very quickly...
displaying stuff very nice...

[shop creating cheap computers]
performing operation not so quickly...
storing data not so quickly...
displaying stuff very poor...
```

## Assignment 2 ('Abstract Factory')

In this assignment computers are created in computershops (again); your task to use Design Pattern 'Abstract Factory' for creating these computers. This means that computer parts (processor, monitor and hard disk) are created through a factory. There are two different factories, a LowBudgetFactory creating cheap computer parts, and a HighBudgetFactory creating expensive computer parts.

Use the following interfaces for the computer parts:



The computer parts are created in method "AssembleMachine" of the ComputerShop class (using a factory), and for each part, the corresponding method is being called (e.g. "PerformOperation" of the processor).

Use the Main code below:

```
static void Main(string[] args)
{
    // create a shop where they assemble expensive computers
    Console.WriteLine("[shop creating expensive computers]");
    // ... create factory
    // ... create shop
    // ... assemble (one) machine

    Console.WriteLine();

    // create a shop where they assemble cheap computers
    Console.WriteLine("[shop creating cheap computers]");
    // ... create factory
    // ... create shop
    // ... assemble (one) machine

    Console.ReadKey();
}
```

to generate the output to the right:

The screenshot shows a console window with the following output:

```
file:///C:/Users/Gerwin van Dijken...
[shop creating expensive computers]
performing operation very quickly...
storing data very quickly...
displaying stuff very nice...

[shop creating cheap computers]
performing operation not so quickly...
storing data not so quickly...
displaying stuff very poor...
```